MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

SUPPLEMENTARY EXAMINATION

TRIMESTER 1, 2015 / 2016

TCP2651 – CONCEPTS OF PROGRAMMING LANGUAGES / TCP2411 – PROGRAMMING LANGUAGE CONCEPTS

(All sections / Groups)

17 NOV 2015
2.30 PM – 4.30 PM
(2 HOURS)

INSTRUCTIONS TO STUDENTS

1. This Question paper consists of **5 pages** only including the cover page with 4 Questions.

2. Attempt **ALL** questions. All questions carry equal marks and the distribution of the marks for each question is given.

3. Please print all your answers **CLEARLY** in the Answer Booklet provided.

Question 1 (5+2+2+6 marks)

(a) Describe and name a programming language for each of the following programming paradigms.

i. Imperative
ii. Declarative
iii. Concurrent
iv. Object-oriented
v. Scripting

(b) How does the support for process abstraction contribute to the writability of a language?

(c) How does the support for data abstraction contribute to the writability of a language?

(d) Draw a figure to illustrate each of the following programming language implementation methods. Then, explain your drawing with an example.

i. Compilation
ii. Pure Interpretation

Continued.......

Question 2 (5+5+2+3 marks)

(a) Consider the following C++ program.

```
#include <iostream>
using namespace std;
int main()
{
  int num = 1 + 2;
  cout << "string" << endl;
}
```

Classify all the lexemes in the program into one of the five tokens. The five tokens are identifier, keyword, operator, literal, and punctuation.

(b) Given the following grammar.

A → I = E
I → a | b | c
E → E + E | E * E | (E) | I

i. Determine from this grammar using derivation whether the sentence "a = b * c + a" is valid or invalid.
ii. Is the grammar ambiguous or unambiguous? Justify your answer.

(c) What is an attribute grammar?

(d) A possible BNF grammar for the assignment statement is as follows:

<assign> → <var> = <expr> ;

i. Write the first semantic rule using attribute grammar so that the data type of the right-hand side of the assignment statement required to match with the left-hand side.
ii. Write the second semantic rule using attribute grammar so that the variable <var> must be declared before it can be referenced in the assignment statement.

Continued.......

Question 3 (2+4+4+2+3 marks)

(a) What are the TWO primary design issues for names?

(b) Provide a suitable example in any programming language to demonstrate each of the following type bindings.

i. explicit static type binding
ii. implicit static type binding
iii. dynamic type binding

(c) The unions in C++ are separate from the records of those language, rather than combined as they are in Ada. What are the advantage and disadvantage of having a separate construct for unions?

(d) Consider the following program.

```
int a = 5;
int fun1() {
      a = 17;
      return 3;
}
void fun2() {
      a = a + fun1();
}
void main() {
      fun2();
}
```

What is the value of "a" assuming:

i. the operands in the expression are evaluated left to right?
ii. the operands in the expression are evaluated right to left?

(e) Assume the C++ rules of associativity and precedence for expressions are followed. Show the order of evaluation by parenthesizing all subexpressions and placing a superscript on the right parenthesis to indicate order for each the following expressions:

i. a + b * c + d
ii. a > b || b++ >5
iii. (a + 1) > 2 | | (a + 1) < -3

Continued.......

Question 4 (2+4+4+2+3 marks)

(a) What are the TWO disadvantages of the goto statement?

(b) Consider the following program.

```
#include <cstdio>
int main() {
    int x = 0;
    do {
        printf("Positive number -> ");
        scanf("%d", &x);
        if(x<=0) goto LabelError;
        printf("good\n");
    } while (x>0);
    LabelError: {
        printf("bad\n");
    }
}
```

Rewrite the above program in C++ without the goto and label statements.

(c) What are the TWO differences between a function and a procedure?

(d) Distinguish between positional and keyword parameters.

(e) Consider the following program written in C syntax.

```
void fun (int a, int b) {
    a += a;
    b += b;
}
void main() {
    int list[2] = {1, 3};
    fun(list[0], list[1]);
}
```

For each of the following parameter-passing methods, what are the values of the list array after execution?

i. Passed by value
ii. Passed by reference
iii. Passed by value-result

End of Page